

Divisible Load Scheduling in Mobile Grid based on Stackelberg Pricing Game

Jiadi Chen, Qiang Zheng, Hang Long, Wenbo Wang

Wireless Signal Processing and Network Lab, Key Laboratory of Universal Wireless Communication,
Ministry of Education, Beijing University of Posts and Telecommunications, Beijing 100876, China
Email: chenjd@bupt.edu.cn

Abstract—Nowadays, it has become feasible to use mobile nodes as contributing entities in computing systems. In this paper, we consider a computational grid in which the mobile devices can share their idle resources to realize parallel processing. The overall computing task can be arbitrarily partitioned into multiple subtasks to be distributed to mobile resource providers (RPs). In this process, the computation load scheduling problem is highlighted. Based on the optimization objective, i.e., minimizing the task makespan, a buyer-seller model in which the task sponsor can inspire the SPs to share their computing resources by paying certain profits, is proposed. The Stackelberg Pricing Game (SPG) is employed to obtain the optimal price and shared resource amount of each SP. Finally, we evaluate the performance of the proposed algorithm by system simulation and the results indicate that the SPG-based load scheduling algorithm can significantly improve the time gain in mobile grid systems.

I. INTRODUCTION

Computational grids and clusters have been widely used to solve computationally-intensive problems [1]. With rapid developments in computer science and manufacturing technologies, many mobile devices, e.g., smart phones, laptop computers, intelligent robots, etc., have growing processing abilities and storage capacities. Together with the advances in wireless communication, it has become feasible to use mobile nodes as contributing entities to grid systems. In another words, mobile devices can act as cloud resources under some circumstances [2].

In this paper, we consider the systems in which mobile nodes can share computing resources with each other through wireless connections. The resource consumer (RC) distribute computational loads to multiple mobile resource providers (RPs) to realize parallel processing. During this process, the load scheduling problem is highlighted. We partition a divisible computing task [3] into multiple subtasks, which have different computing volumes and different input/output data volumes. The load scheduling problem is to determine the partition policy of the overall computational task, then distribute each subtask to the corresponding RP.

Many approaches have been proposed to integrate mobile nodes with grid computing systems, in which the grid schedulers can be categorized by the optimization objectives.

Authors in [4] – [6] try to optimize the makespan for some time sensitive tasks. In [7] – [9], the performance metric is the throughput, which is the key factor to determine the QoS of some certain services, e.g., the data streaming service [10]. There are other approaches that aims at improving the energy efficiency of mobile devices such as [11].

In this paper, we try to minimize the makespan of the overall task by introducing a novel computational load scheduling algorithm into mobile grid systems. Main contributions are summarized as follows:

- i) The reward for sharing idle resources is considered in the problem formulation. Therefore, a buyer-seller model in which the task sponsor can inspire other mobile devices to share their computing resources by paying certain profits, is proposed. The Stackelberg Pricing Game (SPG) [12] is employed to obtain the optimal price and shared amount of each SP's resources.
- ii) Two typical computing scenarios are analyzed, i.e., the proxy-based mobile grid and the mobile ad hoc grid. The RC can be a mobile device or a computing entity in the wired domain, and the RPs can access to a base station/access point or be in a self-organizing way. These two scenarios are highly applicable in the real world systems.
- iii) Simulation results indicate that the proposed SPG-based scheduling algorithm can significantly improve the time gain in mobile grid systems and is proved with good convergence.

The remainder of this paper is organized as follows. The system model is described in Section II. In Section III, the problem formulation and solution is discussed. In Section IV, the simulation results are presented and analyzed. Finally the conclusion is drawn in Section V.

II. SYSTEM MODEL

A. Computing Scenario

As Fig. 1 shows, we consider two computing scenarios in this paper, i.e., the proxy-based mobile grid and the mobile ad hoc grid. In both scenarios, the grid system consists of a variety of mobile devices such as smart phones, laptops, etc., which can act as the resource providers (RPs). The proxy-based mobile grid has one base station (BS) such as a femtocell [13], through which the mobile devices can

¹This work is supported in part by the Fundamental Research Funds for the Central Universities (No.2014ZD03-02), National Key Technology R&D Program of China (2014ZX03003011-004) and China Natural Science Funding (61331009).

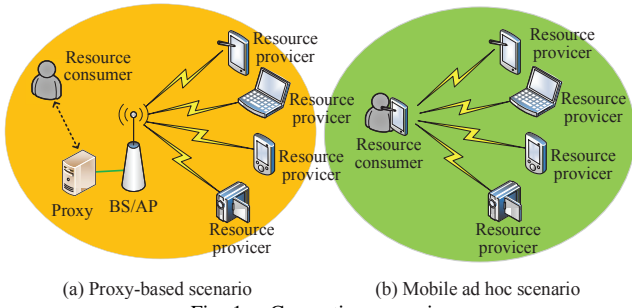


Fig. 1. Computing scenarios.

communicate with each other. While in the mobile ad hoc grid, the RC is a mobile device which distributes the computation load and collect the computation results through device-to-device (D2D) links.

These two scenarios are widely applicable in the real world, based on which the load scheduling problem for mobile grid systems is formulated. Assume that there are K RPs in the mobile grid and each RP has a certain amount of computing resources, e.g., CPU slices. These computing resources can be divided into two parts, i.e., the dedicated computing resource (DCR) and the sharable computing resource (SCR). The former is dedicated for each device to implement basic computing while the latter can be shared for grid computing.

Assume that the RC has certain computing task to perform and it may want to borrow the RPs' SCRs for cooperation. We make the following assumptions:

i) The overall task can be arbitrarily partitioned into multiple subtasks, i.e., computation loads, for multiple RPs to conduct parallel processing. ii) Each subtask has a certain volume of input data which needs to be transmitted to the corresponding RP. There exists a line relationship between the computing volume and input data volume for each subtask; iii) The computing capacity for unit amount of SCR is fixed and the processing time for each subtask is inversely proportional with the occupied SCR amount; iv) The data volume of computing result of each subtask is relatively small so that the result returning delay can be ignored in the problem formulation.

B. System Model

1) *Single-RP scenario*: We first consider the simplest case in which there are only one RC c and one RP v . Denote the input data volume and the computing volume of the overall task as S and V , respectively. The available SCR amount of the RC and RP to process the task are denoted as C_c and C_v . According to assumption (iii) we have $S = \gamma V$, where γ is the balance factor. If the RC perform the overall task by itself, the makespan is $T_l = V/C_c$.

In the parallel processing scenario, we assume that the channel state information (CSI) can be perfectly known by both RC and RP [14]. The maximum achievable data rate of the wireless link between RC (BS/AP) and RP can be given by

$$R_{c,v} = \log(1 + \text{SNR}) = \log\left(1 + \frac{p_c |h_{c,v}|}{\sigma^2}\right) \quad (1)$$

where p_c is the transmission power (in a proxy-based mobile grid, p_c is the power of BS/AP, in the mobile ad hoc grid, p_c is the power of the RC device), which is a constant value since the power control is not considered in this paper. SNR stands for the signal-to-noise ratio. $h_{c,v} \sim \mathcal{CN}(0, 1)$ represents the channel fading coefficient of the link and σ^2 is the additive white noise power at the receiver.

Let β , $0 \leq \beta \leq 1$ denote the proportion of computation load allocated to the RP. To minimize the makespan, the parameter β should satisfy

$$\frac{(1 - \beta)V}{C_c} = \frac{\beta S}{R_{c,v}} + \frac{\beta V}{C_v}. \quad (2)$$

By solving (2), we get

$$\beta^* = \frac{R_{c,v} C_v V}{S C_v C_c + R_{c,v} C_c V + R_{c,v} C_v V}. \quad (3)$$

In this case, the task makespan can be expressed as

$$T_c = T_t + T_p = \frac{\beta^* \gamma V}{R_{c,v}} + \frac{\beta^* V}{C_v}, \quad (4)$$

where T_t denotes the wireless transmission time and T_p is the task processing time. Therefore, the time saved by introducing the parallel processing can be expressed as

$$T_{sv} = \max(T_l - T_c, 0) = \max\left(\frac{V}{C_c} - \frac{\beta^* \gamma V}{R_{c,v}} - \frac{\beta^* V}{C_v}, 0\right). \quad (5)$$

Obviously, T_{sv} is an increasing function of the shared SCR amount C_v . However, the C_v is finite and the RP is also stingy to offer the SCRs unless there are satisfactory returns. So the key issue is how to allocate the computation load and set a reasonable price to meet the optimization objective, i.e., minimizing the makespan of the overall task.

2) *Multi-RP Scenario*: In this part, we extend the single-RP scenario to multi-RP Scenario. Assume that in the mobile grid there are multiple available RPs, the set of which is denoted as \mathcal{A}_c . The maximum data rate that the j -th RP can achieve is expressed as

$$R_{c,j} = \sum_{k=1}^{N_{RB}} s_{k,j} \log\left(1 + \frac{p_c |h_{c,j}|}{\sigma_j^2}\right), \quad j \in \mathcal{A}_c, \quad (6)$$

where $h_{c,j}$ represents the channel gains from RC (or BS/AP) to the j -th RP and σ_j^2 is the additive white noise power at the j -th RP's receiver. N_{RB} denotes the number of transmission resource blocks in the system. In a proxy-based mobile grid, transmission resource blocks can be frequency resource blocks (in FDD system) or time slots (in TDD system). While in a mobile ad hoc grid, the transmission resources blocks can only be time slots for the FDMA is not supported in the D2D communication scene. $s_{k,j} \in \{0, 1\}$ denotes the allocation indicator for the j -th RP at the k -th resource block. $s_{k,j} = 1$ ($s_{k,j} = 0$) indicates that the k -th resource block is (not) allocated to the j -th RP.

Let β_j , $j \in \mathcal{A}_c$ represent the proportion of the overall computation load that be allocated to the j -th RP. β_0 is the

computation load that remains to be processed by RC itself and there has the constraint $\beta_0 + \sum_{j \in \mathcal{A}_c} \beta_j = 1$.

Obviously, β_j must fulfill the same constrain as (2) to minimize the makespan, i.e.,

$$\frac{\beta_0 V}{C_c} = \frac{\beta_j S}{R_{c,j}} + \frac{\beta_j V}{C_j}, \quad j \in \mathcal{A}_c, \quad 0 \leq C_j \leq \bar{C}_j, \quad (7)$$

where C_j is the SCR amount the j -th RP agreed to share and \bar{C}_j is the maximum SCR amount that the j -th RP can provide.

According to the normalization condition $\beta_0 + \sum_{j \in \mathcal{A}_c} \beta_j = 1$ and (7), we can achieve

$$\beta_j = \frac{\beta_0 V R_{c,j} C_j}{S C_c C_j + V C_c R_{c,j}}, \quad (8)$$

$$\beta_0 = \frac{1}{1 + \sum_{j \in \mathcal{A}_c} V R_{c,j} C_j / (S C_c C_j + V C_c R_{c,j})}. \quad (9)$$

Therefore, the time saved in multi-RP scenario can be expressed as

$$T_{mv} = \max(T_l - T_c, 0) = \max\left(\frac{V}{C_c} - \frac{\beta_0 V}{C_c}, 0\right). \quad (10)$$

Similar with (5), the time gain is an increasing function of the occupied SCR amount. The RC makes decision on how much SCRs to “buy” from each SPs based on their “price”. According to assumption (iii), when the shared SCR amount of a certain SP is determined, the computation load distribute to this SP is determined, too. Therefore, the computation load scheduling problem can be solved by determining the price and sold amount of each SPs’ SCRs. In next section, a Stackelberg Pricing Game (SPG) is formulated to highlight this problem and the pricing process is described.

III. PROBLEM FORMULATION

A. Stackelberg Pricing Game

The SPG can be viewed as a seller-buyer interaction game. The sellers, i.e., RPs, wants to maximize their profits via setting a high price. However, the buyer, i.e., RC, attempts to maximize his own utility by lower the price. In a mobile grid, the RC can inspire RPs to share their available SCRs by paying corresponding benefits. Thus, a SPG can be formulated with the RC and RPs as players, and the Stackelberg equilibrium can be considered as the solution of this game. The SPG is a two-level game which can be described from two sides, i.e.,

1) *The Buyer (RC) Side:* The RC wants to acquire the most benefits, i.e., minimize the task makespan, with the least possible cost. As a result, there exists a unique demand/supply-based profile for the RC, given by solving the following optimization problem:

$$\max U_c = T_{mv} - \sum_{j \in \mathcal{A}_c} \lambda_j C_j, \quad s.t. \quad 0 \leq C_j \leq \bar{C}_j, \quad (11)$$

where $\lambda_j C_j$ is the cost paid to the j -th RP, λ_j denotes the price set by the j -th RP for sharing one unit of SCR, and C_j represents the SCR amount that the RC has “bought” from the j -th RP.

2) *The Seller (RP) Side:* The RPs charge the RC for using their SCRs and they attempt to achieve as much profits as possible. Therefore, the SPG at the RPs’ side can be defined as

$$\max U_j = \lambda_j C_j^{b_j} - \eta_j C_j^{b_j}, \quad (12)$$

where $b_j \geq 1$ is a constant tradoff balancer.

Note that the information needs to be exchanged between the RC and RPs are the price and sold amount of each RP’s SCRs. In the following part, the properties and solution of the Stackelberg equilibrium is investigated.

B. Stackelberg Equilibrium

A typical Stackelberg game theoretic equilibrium, in which one player acts as leader and the others as followers, is that the leader sets strategy taking account of the follower’s optimal response. The Stackelberg game will finally converge to the Stackelberg Equilibrium. In this subsection, we give the definition of the Stackelberg Equilibrium and analyze it in a mathematical way.

Definition 1: if the parameters λ_j^{SE} and C_j^{SE} are the Stackelberg Equilibrium of the proposed SPG, the following conditions should be satisfied: when λ_j^{SE} is fixed,

$$U_c(\{C_j^{\text{SE}}\}) = \sup_{0 \leq C_j \leq \bar{C}_j} U_c(\{C_j\}), \quad j \in \mathcal{A}_c, \quad (13)$$

and when C_j^{SE} is fixed,

$$U_j(\{\lambda_j^{\text{SE}}\}) = \sup_{\lambda_i} U_j(\{\lambda_i\}), \quad j \in \mathcal{A}_c. \quad (14)$$

Next we analyze the Stackelberg Equilibrium of our proposed SPG from two aspects, i.e.,

1) *The buyer’s (RC’s) aspect:* As the buyer, the RC is aware of the fact that the sellers, i.e., the RPs, will choose their best response to its strategy. From the previous analysis, the RC maximizes its own utility via buying the optimal volume of SRCs based on the best responses of the followers.

Based on *Definition 1*, differentiate U_c with respect to C_j when λ_j^{SE} is fixed, we can obtain

$$\begin{aligned} \frac{\partial U_c}{\partial C_j} &= -\frac{V}{C_c} \frac{\partial \beta_0}{\partial C_j} - \lambda_j \\ &= \frac{V}{C_c \left(1 + \sum_{j \in \mathcal{A}_c} \frac{V R_{c,j} C_j}{S C_c C_j + V C_c R_{c,j}}\right)^2} \cdot \frac{V^2 C_c R_{c,j}^2}{(S C_c C_j + V C_c R_{c,j})^2} - \lambda_j \\ &= \frac{\beta_0^2 V^3 R_{c,j}^2}{(S C_c C_j + V C_c R_{c,j})^2} - \lambda_j \end{aligned} \quad (15)$$

and

$$\frac{\partial^2 U_c}{\partial C_j^2} = \frac{2\beta_0 \frac{\partial \beta_0}{\partial C_j} V^3 R_{c,j}^2 (S C_c C_j + V C_c R_{c,j}) - 2\beta_0^2 V^3 R_{c,j}^2}{(S C_c C_j + V C_c R_{c,j})^3} < 0. \quad (16)$$

It can be obviously concluded from (15) and (16) that the utility function is concave. Therefore, by setting (15) to zero we can derive the optimal solution C_j^* as

$$C_j^* = u_j \sqrt{1/\lambda_j} - v_j, \quad (17)$$

where

$$\begin{aligned} u_j &= \frac{VR_{c,j}\sqrt{V}}{SC_c + VR_{c,j} + SC_c w_j} \\ v_j &= \frac{VC_c R_{c,j}(1 + w_j)}{SC_c + VR_{c,j} + SC_c w_j} \\ w_j &= \sum_{i \in \mathcal{A}_c, i \neq j} \frac{VR_{c,i}C_i}{SC_c C_i + VC_c R_{c,i}} \end{aligned} \quad (18)$$

Finally, due to the boundary conditions in (11), the optimal amount of SCRs bought from the j -th RP can be expressed as

$$C_j^* = \min(C_j^*, \bar{C}_j) \quad (19)$$

2) *The Seller's (RPs') aspect*: The RPs' goal is to maximize their profits via setting the optimal unit price for the SCR. By differentiating the RPs' utility functions with respect to λ_j and setting it to zero, we get

$$\frac{\partial U_j}{\partial \lambda_j} = (C_j^*)^{b_j} + \lambda_j b_j (C_j^*)^{b_j-1} \frac{\partial C_j^*}{\partial \lambda_j} - \eta_j b_j (C_j^*)^{b_j-1} \frac{\partial C_j^*}{\partial \lambda_j} = 0. \quad (20)$$

By deriving the above equation, we obtain

$$(C_j^*)^{b_j-1} \left(C_j^* + b_j \frac{\partial C_j^*}{\partial \lambda_j} (\lambda_j - \eta_j) \right) = 0 \quad (21)$$

From (21), following conclusion can be achieved: $C_j^* = 0$, which indicates that the consumer won't buy any SCR from the j -th RP. Otherwise,

$$C_j^* + b_j \frac{\partial C_j^*}{\partial \lambda_j} (\lambda_j - \eta_j) = 0, \quad (22)$$

which has a unique solution, i.e.,

$$\lambda_j^* = \eta_j - \frac{C_j^*}{b_j \partial C_j^* / \partial \lambda_j} \quad (23)$$

C. Solution of Stackelberg Equilibrium

In this subsection, a iteration algorithm is employed to obtain the optimal price and shared resource amount. Let $\lambda = \{\lambda_j\}_{j \in \mathcal{A}_c}$ be the price vector and $F_i(\lambda)$ represent the updating function, i.e.,

$$\lambda_j = F_i(\lambda) = \eta_j - \frac{C_j^*}{b_j \partial C_j^* / \partial \lambda_j}. \quad (24)$$

The updating function can be rewritten in a vector form as

$$\lambda(t+1) = \mathbf{F}(\lambda(t)), \quad (25)$$

where $\mathbf{F} = \{F_j\}_{j \in \mathcal{A}_c}$ and t denotes the iteration time. The solution of the SPG can be divided into four steps, which are described in **Algorithm 1**.

IV. SIMULATION RESULTS AND ANALYSIS

We consider the proxy-based mobile grid in the simulation and the scenario is described as follows. A number of mobile devices which act as the RPs, spread over a square room. The RC is a computation node in the wired domain. The wireless AP is located at the center of the room, by which the RC and RPs can communicate with each other. For simplicity, we suppose that the mean SNR from AP to the farthest RP

Algorithm 1 The Solution of SPG

Step 1 - Initialization

For RPs, initialize the price vector $\lambda(0) = \{\lambda_j(0)\}_{j \in \mathcal{A}_c}$ and inform it to the RC. For the RC, initialize its optimal amount of bought SCRs, i.e., $C_j(0)$. Set the iteration index $t = 1$.

Step 2 - RC's update

Use $\lambda(t)$ to obtain the optimal amount of SCRs for each RP according to (17), get $\mathbf{C}(t+1) = \{C_j^*(t+1)\}_{j \in \mathcal{A}_c}$.

Step 3 - RPs' update

Based on $\mathbf{C}(t+1)$ obtained in **Step 2**, calculated the optimal price λ_j^* for each RP according to (24), which are denoted as $\lambda(t+1)$. Then update the load scheduling decision β_j and update the iteration index t by $t+1$.

Step 4

Repeat **Step 2** and **Step 3** until both $\lambda(t)$ and $\mathbf{C}(t)$ do not change any longer or the differential value between two successive iteration is smaller than a predefined threshold. Take the final Stackelberg Equilibrium as the solution to the SPG.

TABLE I
SIMULATION PARAMETERS

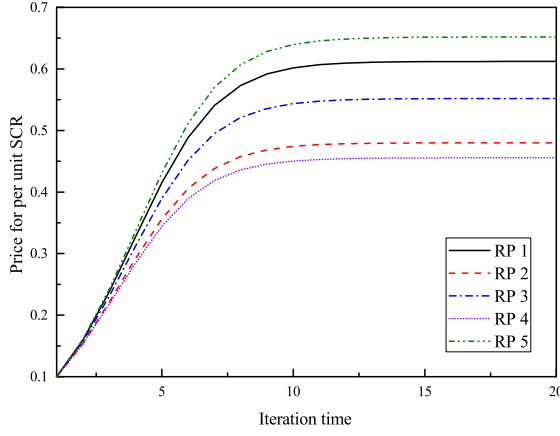
Parameter	Value
Room size	10 m × 10 m
SP number	3 to 12
Distance between devices	3 to 10 m
Pathloss model	$P(d) = Pd^{-\alpha}$
α	4
Bandwidth	10 MHz
Carrier frequency	2 GHz
Scheduling slot length	1 ms

(located at the room edge) equals to 0 dB. The Adaptive Coding and Modulation (AMC) and 16 Modulation and of Coding Schemes (MCSs) [16] are adopted. The channel fading coefficient is i.i.d. over scheduling slots. Main parameter values in the simulation are listed in TABLE I.

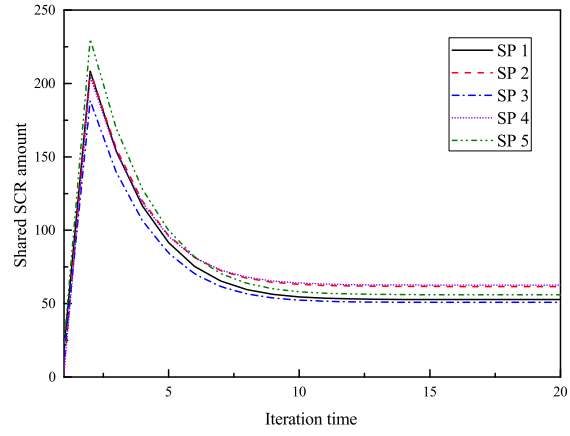
First the convergence performance of the proposed SPG algorithm is evaluated. When there are 5 RPs in the system, Fig. 2 (a) shows the changing process of each SP's price for one unit SCR versus the iteration time. Each SP's SCR price converges to one of the 5 different values with the same initial value 0.1. The higher price is dealt for higher service quality, e.g., higher transmission rate. Similarly, Fig. 2 (b) shows the convergence process of the SCR amount the RC bought from each RP. From Fig. 3 (a) and (b), we can conclude that the proposed algorithm has a good convergence performance, i.e., converges within 15 iteration time under 5-SP scenario.

In the simulation, we evaluate the time gain by realizing the mobile grid computing. Two representative scheduling schemes, i.e., the Round Robin (RR) scheduling and the MaxWeight scheduling [15], are adopted to determine the allocation policy of the wireless transmission resources, i.e., wireless resource blocks. Fig. 4 illustrates the reduction ratio of the task makespan versus the available SP number, and indicates that the adopted wireless resource scheduling scheme can effect the performance of the SPG-based algorithm. Therefore, the proposed computation load scheduling algorithm should cooperate with advanced wireless resource allocation schemes to achieve further performance improvement.

Additionally, we consider another simple and straightforward computation load scheduling policy for comparison, i.e., the SC borrowing equal amount (denoted as a) of SCRs



(a) Optimal price of per unit SCR verse the iteration time.



(b) Optimal SCR amount the RC bought from each RP.

Fig. 2. Convergence performance of the proposed SPG algorithm versus the iteration time.

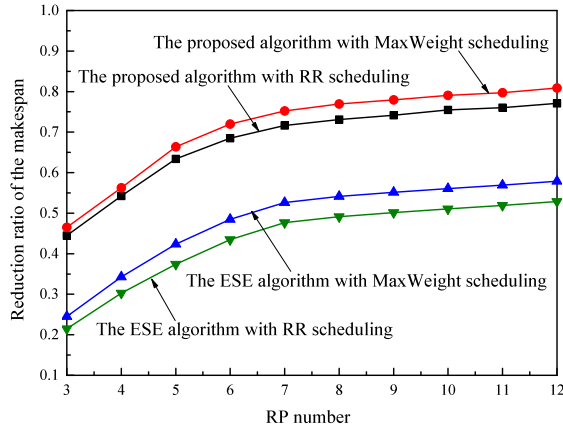


Fig. 3. Proportion of Saving time verse Number of Volunteer.

from each RP. We refer this policy as the ESE algorithm and a is equal to the mean value of the borrowed SCR amount determined by the SPG-based algorithm under same simulation parameters. From Fig. 4 we can see that with the increase of RP number, the time gain by realizing the grid computing is also increasing, and the proposed SPG-based algorithm outperforms the ESE algorithm. When the SP number grows to 12, the ratio of saved time under the proposed algorithm can be up to 80% compared with the makespan in the nonparallel case. Therefore, we can conclude that the proposed computation load scheduling algorithm can significantly improve the computing efficiency in mobile grid systems.

V. CONCLUSION

In this paper, we consider the divisible computation load scheduling problem in grid systems where the mobile devices can share their idle resources for parallel processing. Two main computing scenarios are considered, i.e., the proxy-based mobile grid and the mobile ad hoc grid. The task sponsor want to borrow as much resource as possible to reduce the task makespan, while the resource providers are not willing to share their resources unless there are satisfactory returns. Based on this, a buyer-seller model is designed and a SPG-based algorithm is proposed to obtain the optimal unit price

and shared amount of the computing resources. For performance evaluation, a system simulation is conducted and the results indicate that the proposed SPG-based load scheduling algorithm can significantly improve the computing efficiency in mobile grid systems (80% reduction on the makespan when SP number is 12) as well as a excellent convergence speed (within 15 iteration times when SP number is 5).

REFERENCES

- [1] Mark Baker, Rajkumar Buyya, and Domenico Laforenza, "Grids and grid technologies for wide-area distributed computing," *Software Practice & Experience*, vol. 32, no. 15, pp. 1437-1466, December 2002.
- [2] Lei Lei, Zhangdui Zhong, Kan Zheng, Jiadi Chen, and Hanlin Meng, "Challenges on Wireless Heterogeneous Networks for Mobile Cloud Computing," *IEEE Wireless Communications*, vol. 20, no. 3, pp. 34-44, Jun. 2013.
- [3] Keqin Li, "New Divisible Load Distribution Methods using Pipelined Communication Techniques on Tree and Pyramid Networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 2, pp. 806 - 819, April 2011.
- [4] C. Yu, and D.C. Marinescu, "Algorithms for Divisible Load Scheduling of Data-intensive Applications," *Grid Comput.*, pp.133-155, 2010.
- [5] D.I. Amalarethnam, and F. Selvi, "A Minimum Makespan Grid Workflow Scheduling algorithm," *International Conference on Computer Communication and Informatics (ICCCI)*, pp. 1-6, 2012.
- [6] Rajan, R., and Kamalam, G.K., "Priority based heuristic job scheduling algorithm for the computational grid," *International Conference on Information Communication and Embedded Systems (ICICES)*, pp. 448 - 451, Feb. 2013.
- [7] Sun Bin, and Dong Yueli, "A efficient data replica replacement algorithm for Mobile Grid," *2nd International Conference on Advanced Computer Control (ICACC)*, vol. 1, pp. 510-512, March 2010.
- [8] Morten V. Pedersen, Frank H.P. Fitzek, and Torben Larsen, "Implementation and Performance Evaluation of Network Coding for Cooperative Mobile Devices," *IEEE International Conference on Communications Workshops*, pp. 91 - 96, May 2008.
- [9] Thenmozhi, S., and Tamilarasi, A., "A Cluster Based Resource Allocation Architecture for mobile grid environments," *International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1-5, July 2010.
- [10] S. Muthukrishnan, "Data Streams: Algorithms and Applications," *Foundations & Trends in Theoretical Computer Science*, 2009.
- [11] Chang-Qin Huang, Zhi-Ting Zhu, Yong-He Wu, and Zheng-Hong Xiao, "Power-aware hierarchical scheduling with respect to resource intermittence in Wireless Grids," *International Conference Machine Learning and Cybernetics*, pp. 693-698, 2006.
- [12] Patrick Briest, Martin Hoefer, and Piotr Krysta, "Stackelberg Network Pricing Games," *Computer Science and Game Theory (cs.GT)*, 2008.

- [13] Kan Zheng, Yuyu Wang, Wenbo Wang, and Dohler, M., "Energyefficient Wireless In-home: The Need For Interference-controlled Femtocells," *IEEE Wireless Communications*, vol. 18, no. 6, pp. 36-44, Dec. 2011.
- [14] Kan Zheng, Fei Liu, Lei Lei, and Chuang Lin, "Stochastic Performance Analysis of a Wireless Finite-State Markov Channel," *IEEE Wireless Communications*, vol. 12, no. 2, pp. 782-793, Jul. 2013.
- [15] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, R. Vijayakumar, and P. Whiting, "Scheduling in a queuing system with asynchronously varying service rates," *Probability in the Engineering and Informational Sciences*, vol. 18, pp. 191-217, April 2004.
- [16] 3GPP TS 36.211, V10.4.0, "Technical Specification Group Radio Access Network," Dec. 2011. [Online]. Available: <http://www.3gpp.org>
- [17] Wikipedia, http://en.wikipedia.org/wiki/Stackelberg_game.
- [18] D. Fudenberg, and J. Tirole, "Game Theory," Cambridge: MIT Press, 1993.